AAMC429-USB-22

用户使用手册

V1.0

北京神州航宇测控技术有限公司

目录

手册内容简介.....I

第一章 概述.....1

1.1 功能.....1

1.2 基本工作环境.....1

1.3 安装.....1

第二章 硬件使用说明.....2

2.1 功能说明.....2

2.2 板卡功能结构图.....2

2.3 配置与安装.....2

2.3.1 配置方法.....2

2.3.2 安装步骤.....2

2.4 硬件结构.....2

2.5 硬件通讯接口定义.....3

第三章 驱动接口说明.....4

3.1 功能说明.....4

3.2 运行软件环境描述.....4

3.3 驱动程序文件说明.....4

3.4 安装.....4

3.5 接口说明.....5

3.5.1. 结构与定义.....5

3.5.2. 函数接口描述.....6

3.6 操作建议.....14

第四章 应用程序说明.....16

4.1 软件简介.....16

4.2 运行环境.....16

4.3 安装.....16

4.4 软件界面.....17

4.4.1 程序初始化窗口.....17

4.4.2 板卡号选择窗口.....18

4.4.3 应用程序主窗口.....18

4.4.3 板卡设置窗口.....19

4.5 菜单功能.....19

4.6 操作建议.....19

4.6.1 开启应用程序.....19

4.6.2 板卡号选择.....20

4.6.3 设置板卡.....20

4.6.4 数据通讯操作.....20

4.6.5 退出应用程序.....20

4.7 使用说明.....21

4.7.1 硬件设置.....21

4.7.2 数据的接收与发送.....23
4.8 文件格式.....25
4.9 编程举例.....25
附录A：429 数据格式转换.....27

手册内容简介

本手册主要提供了四个章节，分别为概述、硬件使用说明、驱动接口说明和应用程序说明。

第一章—概述，是对硬件功能与使用方法的整体概述，更多的细节会在第二章—硬件使用说明、第三章—驱动接口说明和第四章—应用程序说明中进行详细描述。

第二章—硬件使用说明，描述了硬件的使用环境、配置安装方法、硬件结构、寄存器描述及用户在实际可能用到的硬件接口。

第三章—驱动接口说明，该部分内容是本手册的重点内容，它详细描述了接口函数的功能及使用方法，通过这一章节的了解，用户可以进行产品的应用程序开发。

第四章—应用程序说明，介绍了提供给用户的应用程序的使用方法，通过现提供的应用程序，用户可以不需要编程而直接进行一些基本的板卡操作，达到快速应用的目的。另外，在该章节中，也提供了驱动接口使用例程，为用户开发适合个人要求的应用程序提供参考。

第一章 概述

1.1 功能

1） USB2.0 总线接口；

2） 2 路发送通道；

3） 2 路接收通道；

4） 具有定时发送功能；

5） 标号过滤功能；

6） 添加时间标签功能；

7） 波特率 100K、48K、12.5K 可设置等功能。

1.2 基本工作环境

操作系统：Windows 98/2000/XP

工作温度：-40 ～ +85℃

相对湿度：0 ～ 95%

1.3 安装

硬件安装：将板卡连接到PC机的USB接口中

驱动安装：见第三章 驱动接口说明

第二章 硬件使用说明

2.1 功能说明

1） USB 总线接口；

2） 2 路发送通道；

3） 2 路接收通道；

4） 具有定时发送功能；

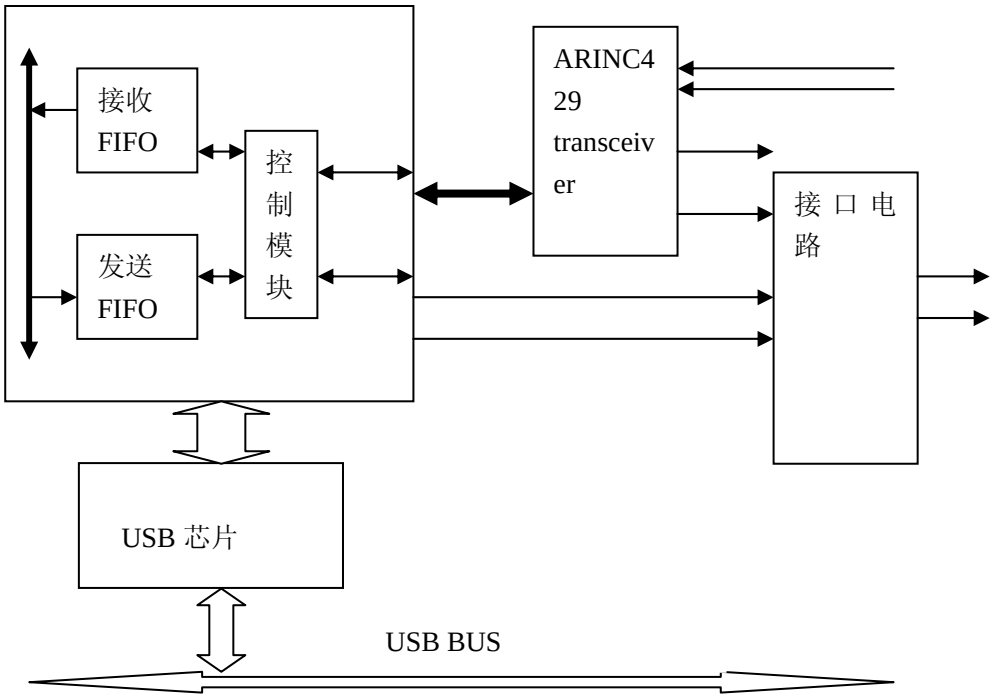
5） 标号过滤功能；

6） 添加时间标签功能；

7） 波特率 100K、48K、12.5K 可设置等功能。

2.2 板卡功能结构图

板卡功能结构图如下所示：



2.3 配置与安装

2.3.1 配置方法

本板卡在使用之前不需要进行硬件配置，所有的初始化都由软件操作

2.3.2 安装步骤

2.4 硬件结构

2.5 硬件通讯接口定义

外部设备通讯连接头：J2 SCSI36 说明如下

管脚	名称	说明	管脚	名称	说明
1	OUT2A	第 2 路发送+	19	OUT2B	第 2 路发送-
2	GND	系统地	20	GND	系统地
3	OUT1A	第 1 路发送+	21	OUT1B	第 1 路发送-
4	GND	系统地	22	GND	系统地
5	IN1A	第 1 路接收+	23	IN1B	第 1 路接收-
6	GND	系统地	24	GND	系统地
7	IN2A	第 2 路接收+	25	IN2B	第 2 路接收-
8	GND	系统地	26	GND	系统地
9			27		
10	GND	系统地	28	GND	系统地
11			29		
12	GND	系统地	30	GND	系统地
13			31		
14	GND	系统地	32	GND	系统地
15			33		
16	GND	系统地	34	GND	系统地
17	GND	系统地	35	GND	系统地
18	GND	系统地	36	GND	系统地

第三章　驱动接口说明

3.1 功能说明

1) USB2.0 的数据接口总线

2) ARINC429 通讯板

3) 2 路接收通道，2 路发送通道

4) 3 种波特率：100K、48K、12.5K

5) 有自检功能，奇偶校验软件可设

6) 支持标号和 SD 过滤功能

7) 支持时间标签功能

8) 支持定时发送功能

3.2 运行软件环境描述

Windows 98/2000/XP。

3.3 驱动程序文件说明

1、库文件：AAMC429_USB.dll 和 AAMC429_USB.lib

2、函数库头文件：AAMC429_USB_lib.h

3.4 安装

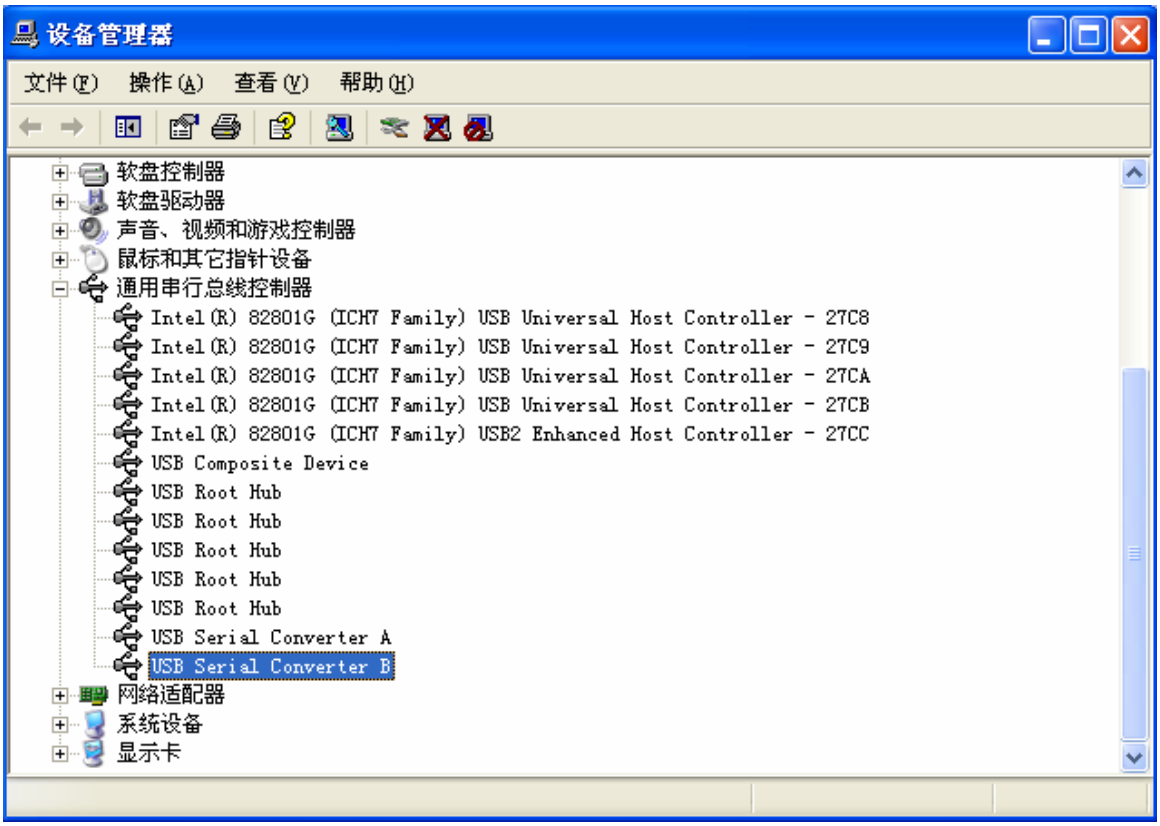
1. 将板卡插入PC机USB接口；

2. 系统提示找到新硬件，需要安装它的软件；

3. 点击硬件安装的下一步，指向光盘中“驱动安装文件”目录，选择ftdibus.inf文件。

4. 点击下一步，直到驱动程序安装成功。

5. 安装完毕，在设备管理器会看到如下信息（蓝色区域）



3.5 接口说明

3.5.1. 结构与定义

3.5.1.1 标号过滤表设置结构

```
typedef struct LabelTable
{
    BYTE LabFilterChan0 [4][256];------第 1 路标号过滤表设置
    BYTE LabFilterChan1 [4][256];------第 2 路标号过滤表设置
} LabelTable_STRUCT;
```

注：LabelFilterChanX：数组的行坐标值代表 S/D 号，列坐标值代表标号；若数组对应元素的值不为零，则 S/D 号对应的标号参与过滤。例：LabelFilterChanX[1][2]=1：表示 S/D 为 1 且 Label 为 2 的 429 数据参与过

滤，即被接收

3.5.1.2 接收包格式

```
typedef struct RevPackage
{
    BOOL TimeTag;//TRUE：时间标签，FALSE：ARINC429 数据
    DWORD RevData;//接收到的时标或 ARINC429 数据
}RevPackage_STRUCT;
```

3.5.1.3 定时时间格式

```
typedef struct TimeInterval
{
    WORD Chan0;//第一路定时时间间隔，单位为 50 微秒
    WORD Chan1;//第二路定时时间间隔，单位为 50 微秒
}TimeInterval_STRUCT;
```

注：设置定时时间间隔时每次设置两路发送通道。

3.5.2. 函数接口描述

3.5.2.1. AAMC429_USB_Open

函数原型：BOOL __stdcall AAMC429_USB_Open (HANDLE *phAAMC429_USB,BYTE CardId);

函数功能：打开 429 板卡，分配板卡资源

参数说明：phAAMC429_USB：板卡的句柄

CardId：板卡编号，当插入 N 块板卡时，板卡号根据插入 PC 机的先后顺序取值分别为 N，N+1,…,(2N-1)，即插入一块板卡时，板卡号取值为 1，当插入两块时，先插入的板卡号为 2，后插入的板卡号为 3。

返回值：　若找到 429 板卡且板卡资源分配成功，返回值为真；否则为假。

3.5.2.2. AAMC429_USB_Close

函数原型：BOOL __stdcall AAMC429_USB_Close(HANDLE hAAMC429_USB);

函数功能：关闭板卡，释放板卡资源

参数说明：hAAMC429_USB：板卡的句柄

返回值： 如果板卡关闭成功，返回值为真；否则为假

3.5.2.3.AAMC429_USB_Init

函数原型：BOOL __stdcall AAMC429_USB_Init(HANDLE hAAMC429_USB, HANDLE hRxEvent,HANDLE hErrEvent);

函数功能：板卡初始化设置，释放板卡资源

参数说明：hAAMC429_USB：板卡的句柄

hRxEvent：接收事件句柄。启动应用程序后当准备接收数据时将此句柄发送给驱动程序，驱动程序开始接收数据。事件句柄由应用程序传递给驱动程序。

hErrEvent：出错事件。当驱动程序出错时，将向应用程序发送此事件。出错条件包括：接收硬件缓冲区溢出，接收软件缓冲区溢出和发送数据出错等。句柄由应用程序传递给驱动程序。

返回值： 如果板卡关闭成功，返回值为真；否则为假

3.5.2.4. ResetCard

函数原型：BOOL __stdcall **ResetCard** (HANDLE hAAMC429_USB);

函数功能：板卡复位函数。停止接收数据；停止四路发送通道的定时发送；停止添加时标；定时发送、标号过滤模式；且板卡处于不通讯状态，此时用户需重新设置波特率。

参数说明：hAAMC429_USB：板卡的句柄

返回值： 若板卡复位成功，返回置为真；否则为假

3.5.2.5. SetConfigureWord

函数原型：BOOL __stdcall **SetConfigureWord** (HANDLE hAAMC429_USB, DWORD CFGWord);

函数功能：设置 429 配置字，包括波特率和数据位长度及发送奇偶校验设置

参数说明：hAAMC429_USB：板卡的句柄

CFGWord：二路发送通道的配置字，结构见下表：

D31	D30	D29	D28	D27	D26	D25	D24
0	0	0	0	0	0	0	0
D23	D22	D21	D20	D19	D18	D17	D16
0	0	0	0	0	0	0	0
D15	D14	D13	D12	D11	D10	D9	D8
cfen1	rclksel12	tclksel1	parity1	R2B10	R2B9	rdec2	R1B10
D7	D6	D5	D4	D3	D2	D1	D0
R1B9	rdec1	sleftest1	paren1	**t2pmod**	**t2pen**	**t2btsel**	**48ken12**

D0：发送 1、2 路，接收 1、2 路波特率 48K 使能　　为 1 时，使能 48K

D1：发送 2 路，波特率选择位　　0　100K　　1　12.5K

D2：发送 2 路奇偶校验位选择　　0　不使能　　1　使能

D3：发送 2 路奇偶校验模式选择位 0　偶校验　　1　奇校验

D4：paren1 第一路发送奇偶校验使能，1 bit31 是校验位，0 bit31 位是数据位

D5：sleftest1 第一路发送、第一路接收和第二路接收自检使能

1 正常工作状态　　0 自检工作状态

在自检状态下，第 1 路发送通道的输出，在内部被连接到第 1、2 路接收通道。其中，数据在进入第 2 路接收通道前被反相，以便与第 1 路接收通道区分。 自检状态下，发送通道仍然正常输出。

D6：rdec1　第 1 路接收通道 S/D 码检测使能。

1： 使能第 1 路接收通道，源/目标 译码器。 0：关闭第 1 路接收通道，源/目标 译码器。

D7、**D8**：R1B9、R1B10　第 1 路接收通道 S/D 码。

如果 rdec1=1，第 1 路接收通道将收到 ARINC429 数据的 S/D 与 R1B9,R1B10 进行比较，相同则接纳，不相同则丢弃。其中，R1B9 与串行数据 bit9 比较，R1B10

与串行数据 bit10 比较。

D9：rdec2　第 2 路接收通道 S/D 码检测使能。

1：　使能第 1 路接收通道，源/目标　译码器。　0：关闭第 1 路接收通道，源/目标　译码器。

D10、**D11**：R2B9、R2B10　第 2 路接收通道 S/D 码。

如果 rdec2=1，第 1 路接收通道将收到 ARINC429 数据的 S/D 与 R2B9,R2B10 进行比较，相同则接纳，不相同则丢弃。其中，R2B9 与串行数据 bit9 比较，R2B10 与串行数据 bit10 比较。

D12：parity1　发送 1 路校验模式选择

0 奇校验　1 偶校验

D13：tclksel1 发送 1 路波特率模式选择

0 100k　1 12.5k

D14：rclksel12 接收 1 路、2 路波特率模式选择

0 100k　1 12.5k

D15：配置使能

配置接收 1、2 路，发送 1、2 路

D16-D31：恒设为 0

接收恒定采用奇校验

3.5.2.6. SetLabelFilter

函数原型：void __stdcall **SetLabelFilter** (HANDLE hAAMC429_USB, LabelTable_STRUCT *Label);

函数功能：设置标号过滤表，即某种标号和 S/D 码的 429 数据将被接收

参数说明：hAAMC429_USB：板卡的句柄

Label：指向标号过滤表设置结构的指针

返回值：　空

3.5.2.7. StartLabelFilter

函数原型：BOOL __stdcall **StartLabelFilter** (HANDLE hAAMC429_USB, BYTE ChannelNo, BOOL Enable);

函数功能：接收通道 429 数据的标号和 S/D 码参与过滤使能函数

参数说明：hAAMC429_USB：板卡的句柄

ChannelNo：接收通道号，取值范围为 0-1，分别对应接收通道 1-2

Enable：使能位；

TRUE：接收通道将只接收标号和 S/D 码符合标号过滤表中的 429 数据

FALSE：接收通道将接收所有标号和 S/D 码的 429 数据

返回值：若标号过滤功能使能成功，返回值为真；否则为假

3.5.2.8. AddTimeTag

函数原型：BOOL __stdcall **AddTimeTag** (HANDLE hAAMC429_USB, BYTE ChannelNo, BOOL Enable);

函数功能：设置时间标签模式

参数说明：hAAMC429_USB：板卡的句柄

ChannelNo：接收通道号，取值范围为 0-1，分别对应接收通道 1-2

Enable：模式使能位；

TRUE：使能，接收到的 429 数据将带一个时间标签，表明数据接收到的时间

FALSE：不使能

返回值：若时标模式设置成功，返回值为真；否则为假

3.5.2.9. StartTimeTag

函数原型：BOOL __stdcall **StartTimeTag** (HANDLE hAAMC429_USB);

函数功能：将系统时间写入硬件

参数说明：hAAMC429_USB：板卡的句柄

返回值：若操作成功，返回值为真，否则为假

3.5.2.10. SetTimerInterval

函数原型：BOOL __stdcall **SetTimerInterval**(HANDLE hAAMC429_USB,

BYTE ChannelNo, TimeInterval_STRUCT *Time);

函数功能：设置群定时发送时间间隔。**定时发送的时间间隔为两次启动定时发送数据的起始时间的时间差；定时发送的时间应该保证发送完所有的数据，否则发送的数据会不全。**

参数说明：hAAMC429_USB：板卡的句柄

ChannelNo：通道号，取值为 0，设置发送 1、2 路定时时间

Time：指向定时时间结构的指针

返回值：若定时时间设置成功，返回值为真；否则为假

3.5.2.11. StartTimer

函数原型：BOOL __stdcall **StartTimer**(HANDLE hAAMC429_USB,

BOOL Enable);

函数功能：启动或停止定时发送

参数说明：hAAMC429_USB：板卡的句柄

Enable：使能位；

TRUE：若设置了定时发送时间，硬件将按定时时间间隔重复发送 FIFO 里的数据

FALSE：硬件将停止数据的发送

返回值：若定时器启动成功，返回值为真；否则为假

3.5.2.12. **ReadRevFIFOStatus**

函数原型：DLL BYTE __stdcall ReadRevFIFOStatus(

HANDLE hAAMC429_USB, BYTE ChannelNo);

函数功能：读取接收 FIFO 的状态，包括接收 FIFO 的空、满、不满标志

参数说明：hAAMC429_USB：板卡的句柄

ChannelNo：接收通道号，取值范围为 0-1，分别对应接收通道 1-2

返回值： 返回值见下表：

二进制值	符号名	意义
0x11	FIFOEmpty	FIFO 空
0x12	FIFOFull	FIFO 满

0x13	FIFONotFull	FIFO 不满 ， 但也不空

3.5.2.13. ReadFIFOStatus_S

函数原型：DLL BYTE __stdcall ReadTranFIFOStatus(

HANDLE hAAMC429_USB)

函数功能：读取 USB 发送 FIFO 的状态，包括发送 FIFO 的满、不满标志

参数说明：hAAMC429_USB：板卡的句柄

返回值： 返回值见下表

二进制值	符号名	意义
0x12	FIFOFull	FIFO 满
0x13	FIFONotFull	FIFO 不满 ， 但也不空

3.5.2.14. **CountsInRevFIFO**

函数原型：DLL WORD __stdcall CountsInRevFIFO(

HANDLE hAAMC429_USB, BYTE ChannelNo)

函数功能：读取接收缓冲区中数据个数

参数说明：hAAMC429_USB：板卡的句柄

ChannelNo：待接收数据的通道号，取值范围为 0-1；分别对应接收通道 1-2

返回值：对应接收通道接收缓冲区中数据个数

3.5.2.15. **ReadErrorStatus**

函数原型：DLL BYTE __stdcall ReadErrorStatus(HANDLE hAAMC429_USB);

函数功能：读取错误状态

参数说明：hAAMC429_USB：板卡的句柄

返回值：返回值见下表

二进制值	符号名	意义
0x20	RxOverflow1	硬件第一路接收 FIFO 溢出
0x21	RxOverflow2	硬件第二路接收 FIFO 溢出
0x24	TxOverflow1	硬件第一路发送 FIFO 溢出
0x25	TxOverflow2	硬件第二路发送 FIFO 溢出
0x28	RxBufOverflow1	软件第一路接收缓冲区溢出
0x29	RxBufOverflow2	软件第二路接收缓冲区溢出

3.5.2.16. ReceiveData_Dword

函数原型：DLL void __stdcall ReceiveData_Dword(HANDLE hAAMC429_USB,BYTE ChannelNo, RevPackage_STRUCT * Buffer);

函数功能：接收数据函数

参数说明：hAAMC429_USB：板卡的句柄

ChannelNo：待接收数据的通道号，取值范围为 0-1，分别对应接收通道 1-2

Buffer：指向 RevPackage_STRUCT 结构的指针

返回值：无

3.5.2.17. ReceiveData_Block

函数原型：DLL BOOL __stdcall ReceiveData_Block(HANDLE hAAMC429_USB,BYTE ChannelNo, RevPackage_STRUCT * buf, WORD dwDwords);

函数功能：使能数据接收

参数说明：hAAMC429_USB：板卡的句柄

ChannelNo：待接收数据的通道号，取值范围为 0-1，分别对应接收通道 1-2

buf：指向 RevPackage_STRUCT 结构的指针

dwDwords：待接收数据个数

返回值：若接收成功返回真，否则返回假

3.5.2.18. SendData

函数原型：BOOL __stdcall SendData (HANDLE hAAMC429_USB, BYTE ChannelNo, WORD Len, DWORD *DataAry);

函数功能：数据发送函数

参数说明：hAAMC429_USB：板卡的句柄

ChannelNo：待发送数据的通道号，取值范围为 0-1，分别对应发送通道 1-2

Len：待发送数据的长度

DataAry：存放待发送的数据

返回值：若数据发送成功，返回值为真；否则为假

3.6 操作建议

1. 创建接收事件和错误监视事件
2. 打开板卡（AAMC429_USB_Open）
3. 复位板卡（ResetCard）
4. 启动时标计数器（StartTimeTag）
5. 设置配置字（SetConfigureWord）
6. 设置标号过滤功能
   1） 设置待接收数据的标号（SetLabelFilter）
   2） 启动标号过滤功能（StartLabelFilter）
7. 设置时间标签（AddTimeTag）
8. 定时发送功能设置
   1） 设置定时发送的时间间隔（SetTimerInterval）
   2） 启动定时发送模式（StartTimer）
9. 初始化设置（AAMC429_USB_Init）

   创建接收事件和错误监视事件，并把事件句柄传给驱动程序。
10. 监视错误状态
    1） 创建错误监视线程
    2） 等待驱动程序在发现错误时唤醒线程
    3） 读取错误状态（ReadErrorStatus），进行相应的处理
11. 数据的接收

    在准备接收数据前，须先唤醒接收线程。

- 按字方式读取数据
  1）判断接收 FIFO 是否为空（ReadRevFIFOStatus）
  2）若不为空（不为 FIFOEmpty），读取数据（ReceiveData_Dword）
  3）重复以上 2 步

- 按块方式读取数据

  1）读取接收缓冲区中数据的数量（CountsInRevFIFO）

  2）读取一块数据（ReceiveData_Block）

  注：若添加时标时读取数据的个数必须为偶数。

说明：若设置了时间标签模式，则接收数据时，首先收到的是一个 429 数据（记为 data429），接下来的是该 429 数据的时标（记为 TimeTag）。

12. 数据的发送

- 使用定时发送

  1）停止定时器（StartTimer）

  2）启动定时器（StartTimer）

  3）读取发送 FIFO 的状态（ReadFIFOStatus_S）

  4）若发送 FIFO 不满（为 FIFONotFull）且没有溢出（不为 FIFOError），则将数据写到发送 FIFO（SendData）

- 不使用定时发送

  1）读取发送 FIFO 的状态（ReadFIFOStatus_S）

  2）若发送 FIFO 不满（为 FIFONotFull），则发送数据（SendData）

  3）重复以上 2 步

13. 应用程序退出时，关闭板卡（AAMC429_USB_Close）

第四章 应用程序说明

4.1 软件简介

AAMC429-USB-22 应用程序是本公司针对 USB 接口二收二发 429 板卡产品开发的应用程序，目的是为了使用户很快地对硬件的功能有所了解。

本应用程序通过调用驱动接口程序，可以对硬件板卡提供的所有功能进行设置，还可以进行简单的数据通讯操作。

4.2 运行环境

硬件需求：

可支持 USB 总线的计算机

至少 128M 内存

显示器分辨率最低可设为 800*600

AAMC429-USB-22 通讯板

软件需求：

Windows 98/2000/XP 操作系统

USB 硬件驱动程序

4.3 安装

执行光盘中所带有的应用程序安装包 AAMC429-USB-22.exe，按照安装提示进行程序安装，完成安装后，用户可通过执行任务栏中的“开始”->“程序”->“AAMC429-USB-22”->“AAMC429-USB-22”来运行应用程序。

4.4 软件界面

4.4.1 程序初始化窗口

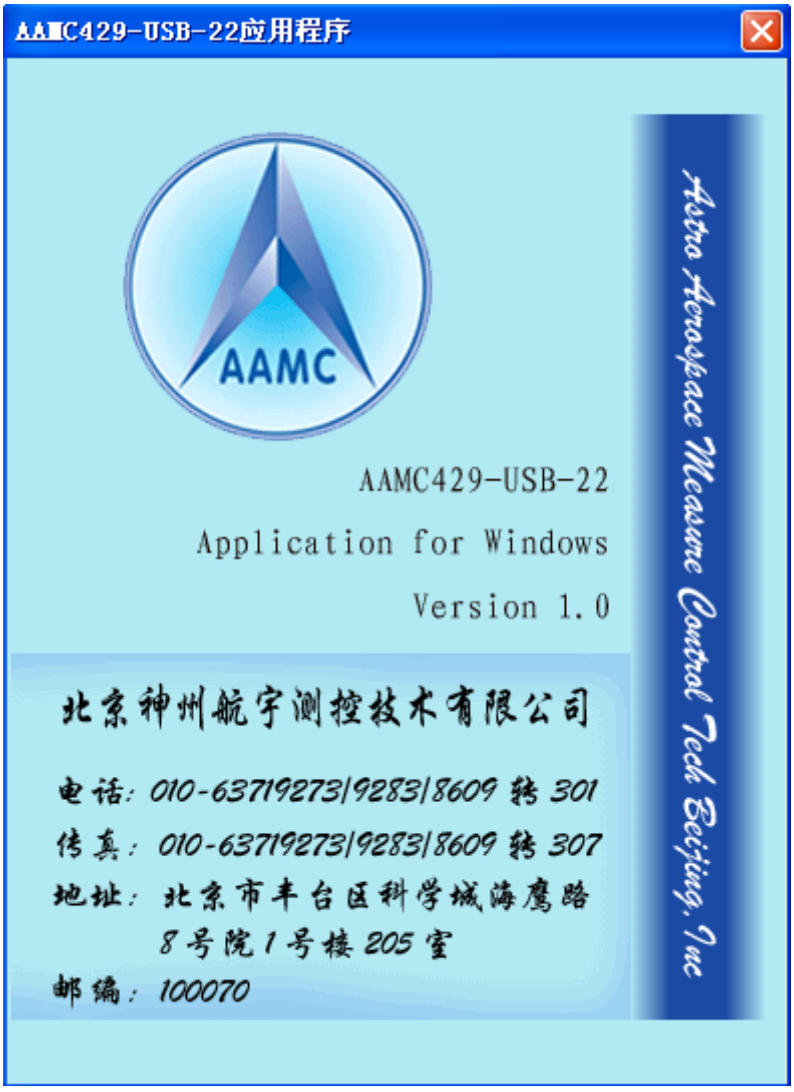


图 I-1（程序初始化窗口）

该窗口在三秒后自动关闭，也可点击“关闭”按钮关闭。

4.4.2 板卡号选择窗口

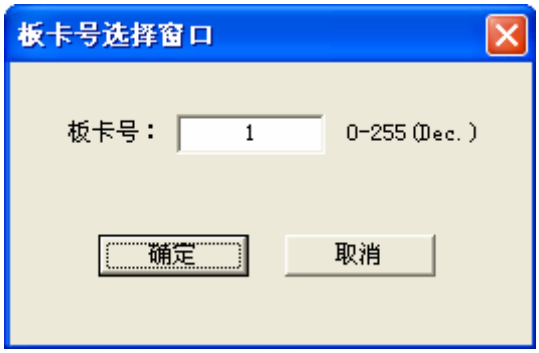


图 I-2（板卡号选择窗口）

4.4.3 应用程序主窗口

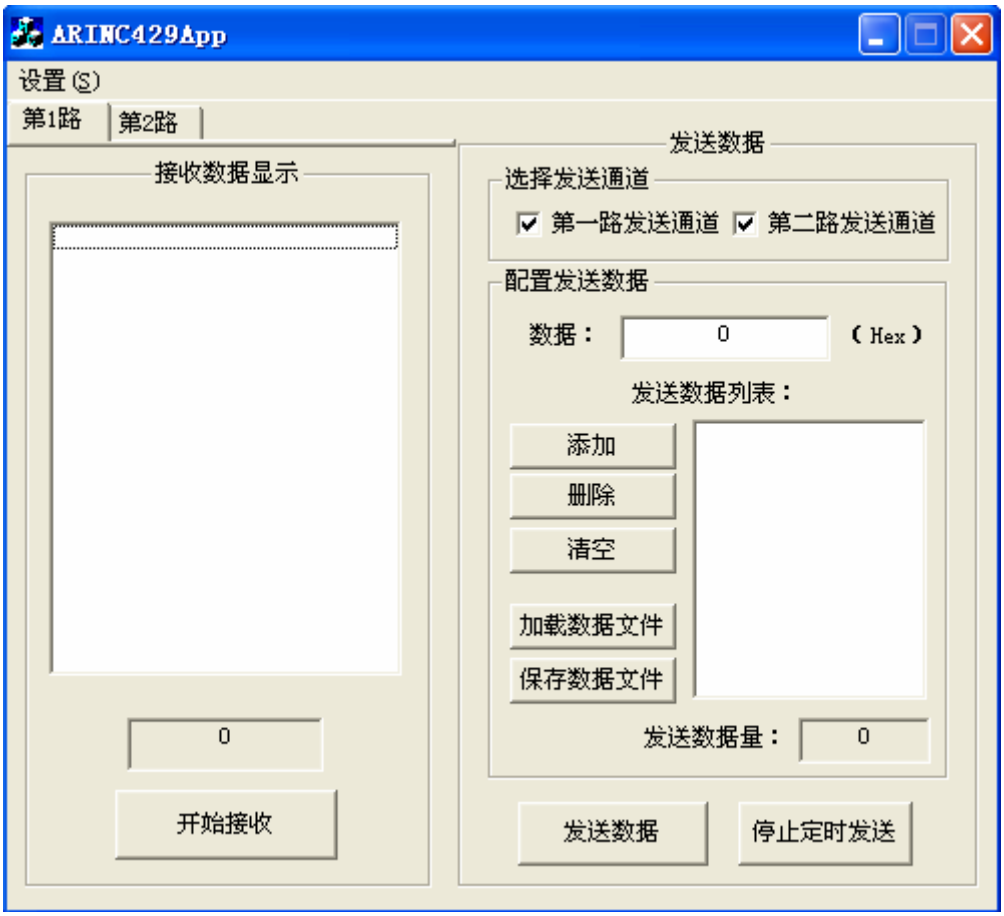


图 I-3（主窗口）

4.4.3 板卡设置窗口

图 I-4（设置窗口）

4.5 菜单功能

设置----------显示板卡设置窗口（图 I-4 所示）

4.6 操作建议

4.6.1 开启应用程序

用户可通过执行任务栏中的“开始”->“程序”->“AAMC429-USB-22”->“AAMC429-USB-22”来运行应用程序。

4.6.2 板卡号选择

应用程序启动后，需要做的第一件事就是进行板卡号选择（如图 I-2 所示）。当用户的计算机中同时插有多个同类型的板卡时，“选择板卡号”就用来指定用户当前操作的是哪块板卡,它的输入范围是从 1 到 127 的十进制数。（当插入 N 块板卡时，板卡号根据插入 PC 机的先后顺序取值分别为 N，N+1,…,(2N-1)）。

如果指定序号的板卡存在，则直接进入主窗口（如图 I-3）。

4.6.3 设置板卡

在进入主窗口后，程序会自动对板卡进行默认配置，该配置结果允许用户可以进行最基本的数据发送与接收操作，即无触发深度、无定时发送、无时间标签，也没有标号过滤。要进行更多的功能配置，还需要进一步配置板卡。

执行菜单“设置”，程序会显示设置窗口（如图 I-4 所示），这个窗口中包含了硬件所有可被设置的属性。

完成设置后，单击“确定”按钮向板卡提交设置，也可以单击“取消”结束本次操作。

4.6.4 数据通讯操作

应用程序提供了简单的数据通讯操作，如数据的接收与发送。程序会将收到的数据显示在接收数据列表中。用户可以通过窗口界面配置发送数据，还可以从文件中加载发送数据，并通过软件将数据写入发送缓冲区，再由硬件发送出去。

4.6.5 退出应用程序

单击“主窗口”右上角的“X”按钮，便可退出应用程序，即出程序时有提示，确认后即可退出。

4.7 使用说明

4.7.1 硬件设置

在应用程序开启后，用户可以通过执行主窗口菜单中“设置”，显示硬件板卡设置对话窗口（如图 I-4 所示）。

4.7.1.1 设置配置字

配置字包括：比特率和校验，如下表所示：

配置字内容	说明		
比特率	100K	48K	12.5K
校验	奇校验	偶校验	无

第 1 路发送与第 1、2 路接收通道均采用相同的配置字，在设置窗口中共有 2 个配置卡，按这种规则，配置字对发送与接收通道的影响关系如下表所示：

发送通道	影响关系
第 1 路发送	第 1、2 路接收与第 1 路发送
第 2 路发送	第 2 路发送

修改任意一个配置卡中的配置字，与其存在相互影响关系的另一个配置卡中的配置字也会随之改变，软件会自动确保配置字在设置时保持一致。

4.7.1.2 设置定时发送

定时发送是指将缓冲区内的数据按指定的时间间隔发送出去。本程序提供了对四路发送通道的定时发送功能的设置。

定时时间的基本单位为 50 微秒，当输入值为 20000 时，则表示实际定时发送的时间间隔为 1 秒。具体公式如下：

实际时间间隔 = 时间间隔输入值 × 单位时间

经过如上公式，我们可以算出实际的时间间隔值：

实际时间间隔 = 20000 × 50

= 1000000（微秒）= 1（秒）

定时发送值设置为 0 时，则表示不采用定时发送功能。

4.7.1.3 添加时间标签

时间标签是指在接收到的数据后面增加一个接收到的时间，这个时间就称为

时间标签，该标签的作用是用来描述当前接收到的这个数据的到达时间。其单位是 50 微秒，它的计时完全由硬件来实现。

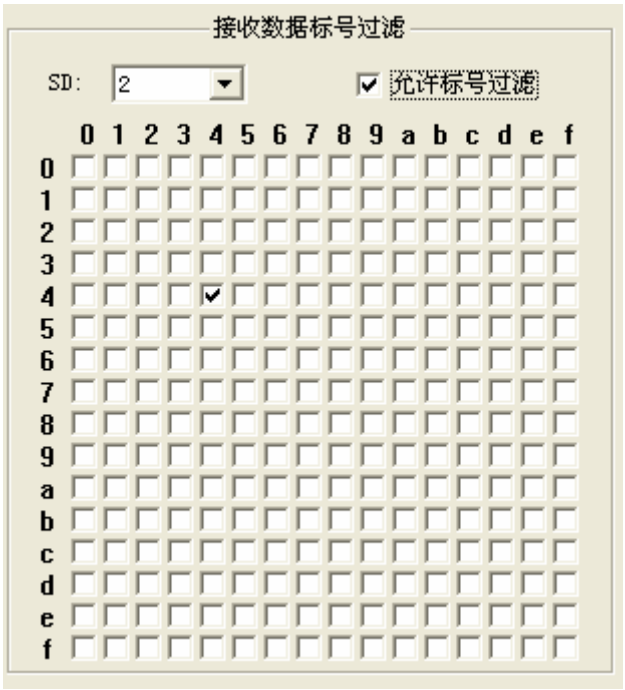
4.7.1.4 设置接收标号过滤

标号过滤实际上就是，通知硬件只接收那些用户想收到的数据。该功能的实现主要是依据 429 标准，通过识别数据的标号实现数据过滤功能，标号指的是 ARINC429 数据中的第 0 位到第 7 位表求的一个 8 位二进制数据。

当数据位设置为 32 位时，SD 值也参与过滤，SD 值指 ARINC429 数据中的第 11、12 两位，SD 下拉列表中的数据为 0 到 3，描述二进制数据的两个数据位。

当数据位设置为 25 位时，SD 值则不参与过滤，软件中每一个 SD 值都对应着 256 个标号。设置标号过滤时，与 32 位的有些区别，需要将 4 个 SD 值对应的 4 组标号均设置为相同的，而在 32 位数据通讯时，由于 SD 参与了过滤，则每个 SD 值都可以对应不同的标号设置。

在接收标号过滤设置中，共有 256 个标号可设置，分别是从 0 到 255。且每一个 SD 值都有 256 个标号。标号里横向的是低 4 位，纵向的是高 4 位，如果用户要设置一个 SD 为 2 标号为 44H（68D）的数据过滤时，需要先将 SD 选择在 2 上，然后再在相应的标号上选中即可，操作结果如下所示：



在设置 256 个标号时当用户选中了其中的某一项，就表示对该标号的数据进行过滤，也就是会接收到该标号的数据。

单配置了标号与 SD 是不够的，只有将“允许标号过滤”选中，才可以便板卡具有标号过滤功能。

4.7.1.5 板卡工作方式

方式	说明
自环	在该方式下，板卡实现内部自环功能，主要表现为，板卡可以在不与外界通讯时，便可接收到自己发送的数据，AAMC429-USB-22 板卡具有 2 路接收与 2 路发送的能力，通过本程序可以从 2 路接收通道接收到发送的数据，但需要注意的是，此时奇数序列接收通道收到的是原数据，偶数序列接收通道接收到的数据为原数据的取反值。该工作方式下也存在和配置字设置同样的对应关系，例如，从第 1 路发送的数据会由 1、2 路接收收到。
正常	在该方式下，板卡可保证正常通讯。

4.7.1.6 配置的保存与加载

在每次应用程序打开后，程序会自动按照默认值对板卡进行配置，实现基本的数据通讯能力，但往往用户都需要将板卡的功能配置成自己想要的功能，这样就需要经常对板卡进行配置操作，在设置窗口中的“加载配置”与“保存配置”可以提供些方便。当用户进入到配置窗口后，首次对板卡通讯进行自定义配置，配置完成后，可以执行“保存配置”，这时程序会将程序中当前的配置保存到配置文件中，以便下次重新启动应用程序后，用户可以通过执行“加载配置”操作重新获取之前保存的配置。

4.7.2 数据的接收与发送

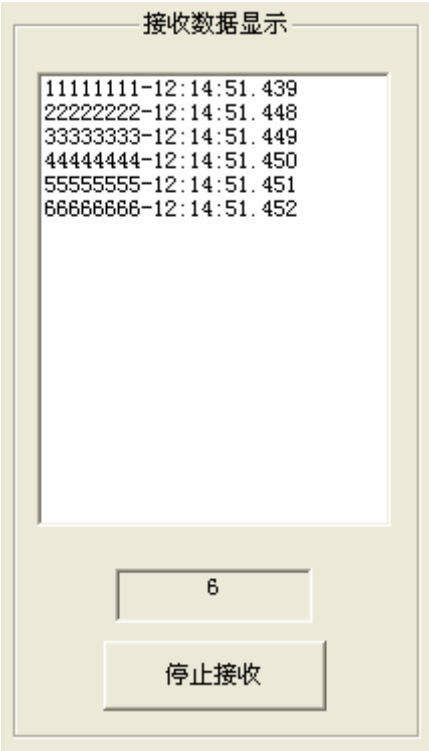
该板卡有 2 路接收和 2 路发送通道，主窗口用于实现数据通讯操作（如图 I-3 所示）。

4.7.2.1 接收数据

需要进行数据接收时，执行“开始接收”就可以了，执行它后，该按钮会自动变为“停止接收”按钮，终止接收操作时，只需再次点击一下该按钮就可以了。

程序会将接收到的数据分别显示在列表窗口中，接收到的数据量会显示在列表窗口下方的文本窗口中。查看接收数据可以通过选择接收通道卡实现。

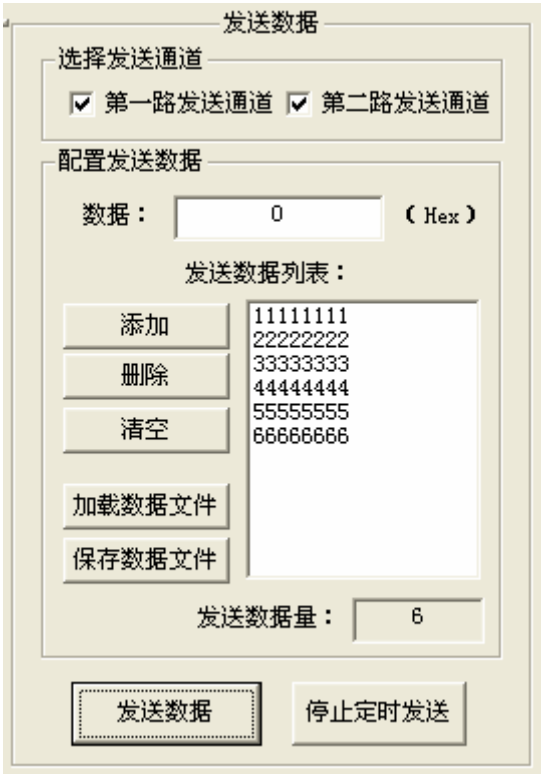
接收数据时，如果添加了时标功能，在接收列表中的数据显示会有所变化。在每一个数据后面都会加上“时:分:秒.毫秒”的显示，用来表示当前数据接收到的时间。如下图所示：



4.7.2.2 发送数据

发送数据时可以通过执行“发送数据”进行数据发送，如果配置了定时发送功能，则可通过执行“停止定时发送”停止定时发送。

程序总是会将发送列表中的数据发送出去，所示要保证发送数据列表不为空。添加发送数据到列表有两个方法，一个就是通过“添加”、“删除”、“清空”操作将数据文本窗口中的数据添加到列表，输入数据时要以十六进制的输入方式进行输入，列表中显示的数据也都是十六进制的。另一个方法是通过“加载数据文件”和“保存数据文件”进行发送列表的配置，“加载数据文件”会指定一个数据文件，并将里面的数据以续加的方式添加到列表中，不会将列表中原有的数据清空。“保存数据文件”会将列表中的数据保存到文件中。两种配置发送数据列表的方法可以混合使用。



注：“删除”操作会将发送列表中被选中的数据删除掉，当列表中没有数据被选中时，将会从列表中的最后一个数据开始删除。

4.8 文件格式

当完成安装后，在安装目录中应有如下文件：

ARINC429App.exe 应用程序

AAMC429_USB.dll 驱动程序接口

AAMC429_USB.lib 驱动程序库文件

aamc429cfg.ini 配置文件，用于保存用户对板卡的配置

data.sdt 数据文件，可由用户自定义修改，里面存放发送数据。

注：除数据文件可以由用户通过软件修改，其它的三个文件不可以由用户自行修改。

4.9 编程举例

为方便开发个性化的应用程序，我公司提供了应用程序例程，该例程可从交付客户的光盘中获得。例程采用的是 VC 6.0 工具进行编写，所用的编程语言是

C 语言。这里我们仅是提供了一个简单的例子，但我们的动态链接库是可以被 Visual C++、Visual Basic、Delphi、LabView 以及 CVI 等开发工具调用的。

附录 A：429 数据格式转换

图上面的为 429 总线上的数据格式，而下面的为计算机系统软件的数据格式。即发送 FIFO 发送的数据与 429 串行数据总线上的数据进行下面的格式转换。其中 Word1 是系统数据低 16 位，而 Word2 是系统数据高 16 位。

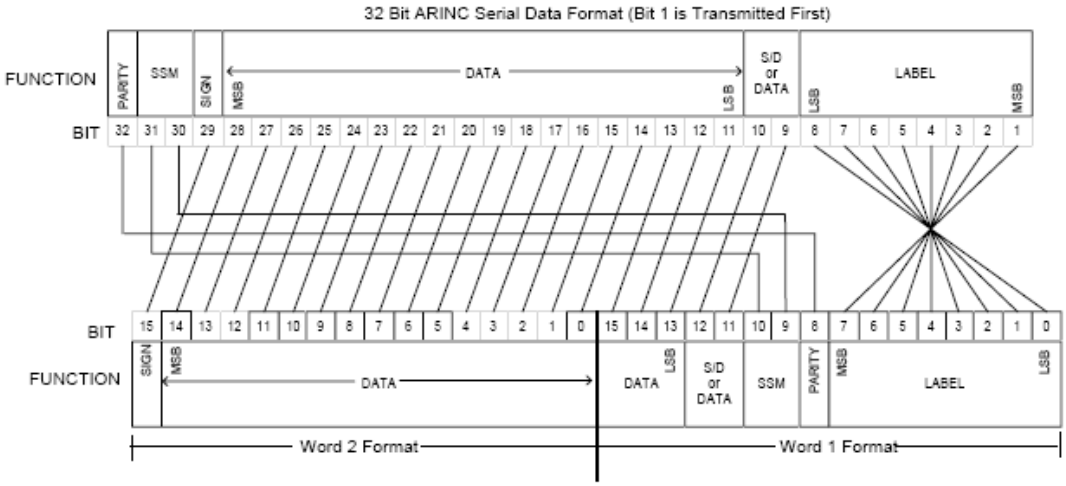


Figure 2: Mapping of Serial Data to/from Word 1 and Word 2 in 32 bit format.

例如：系统软件写数据 12345678 到发送 FIFO，奇偶校验设置位奇。则系统数据为

Word1=0X5678，Word2=0X1234。

429 串行数据格式为：0X62468A1E